

Attorney Docket No. 122.1560

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Patent Application of:

Hiroshi DAIKU, et al.

Application No.:

Group Art Unit:

Filed: July 3, 2003

Examiner:

For: SOLID-STATE IMAGE SENSOR

**SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN  
APPLICATION IN ACCORDANCE  
WITH THE REQUIREMENTS OF 37 C.F.R. § 1.55**

Commissioner for Patents  
PO Box 1450  
Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 37 C.F.R. § 1.55, the applicant(s) submit(s) herewith a certified copy of the following foreign application:

Japanese Patent Application No(s). 2002-216848 and 2002-316280

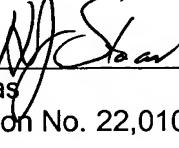
Filed: July 25, 2002 and October 30, 2002

It is respectfully requested that the applicant(s) be given the benefit of the foreign filing date(s) as evidenced by the certified papers attached hereto, in accordance with the requirements of 35 U.S.C. § 119.

Respectfully submitted,

STAAS & HALSEY LLP

Date: July 3, 2003

By:   
H. J. Staas  
Registration No. 22,010

1201 New York Ave, N.W., Suite 700  
Washington, D.C. 20005  
Telephone: (202) 434-1500  
Facsimile: (202) 434-1501

日本国特許庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されて  
いる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed  
with this Office

出願年月日

Date of Application: 2002年 7月25日

出願番号

Application Number: 特願2002-216848

[ ST.10/C ]:

[JP2002-216848]

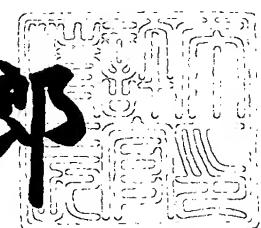
出願人

Applicant(s): 富士通株式会社

2003年 5月13日

特許庁長官  
Commissioner,  
Japan Patent Office

太田信一郎



出証番号 出証特2003-3034407

【書類名】 特許願

【整理番号】 0241095

【提出日】 平成14年 7月25日

【あて先】 特許庁長官殿

【国際特許分類】 H04N 5/335

【発明の名称】 CMOSイメージセンサ

【請求項の数】 7

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通  
株式会社内

【氏名】 國分 政利

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通  
株式会社内

【氏名】 水口 寿孝

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通  
株式会社内

【氏名】 船越 純

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通  
株式会社内

【氏名】 小林 博

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通  
株式会社内

【氏名】 山本 克義

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通

株式会社内

【氏名】 西尾 茂

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通  
株式会社内

【氏名】 大工 博

【特許出願人】

【識別番号】 000005223

【氏名又は名称】 富士通株式会社

【代理人】

【識別番号】 100108187

【弁理士】

【氏名又は名称】 横山 淳一

【電話番号】 044-754-3035

【手数料の表示】

【予納台帳番号】 011280

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0017694

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 CMOSイメージセンサ

【特許請求の範囲】

【請求項1】

フォトダイオードと、リセットトランジスタと、増幅トランジスタと、水平方向選択トランジスタを備えたCMOSイメージセンサであって、

リセットノイズの読み出し信号を監視し、所定の判定電圧に応じて信号をリセット電位に固定する検出回路を備えたことを特徴とするCMOSイメージセンサ。

【請求項2】

画素部がアレイ状に配置されてなるCMOSイメージセンサであって、  
着目画素と縦方向又は横方向に隣接する両隣の同色画素の画素データの平均値  
に所定の不感帯を付加することにより所定の欠陥判定範囲に設定し、該欠陥判定  
範囲内に該着目画素の画素データが含まれるときは、該着目画素を正常の画素と  
判定する判定回路を備えたことを特徴とするCMOSイメージセンサ。

【請求項3】

画素部がアレイ状に配置されてなるCMOSイメージセンサであって、  
カラーフィルターの色成分よりも多い数のレジスタを有し、それぞれのレジス  
タによりコラムごとに独立に所定のゲインとオフセットを設定するカラー処理回  
路を備えたことを特徴とするCMOSイメージセンサ。

【請求項4】

画素部がアレイ状に配置されてなるCMOSイメージセンサであって、  
画像縮小後の仮想画素を想定し、該仮想画素を目標にして、該仮想画素の周辺  
の画素の画素データを、カラーフィルターの各色の位置情報をできるだけ同じに  
なるような重み付け計算を行って、間引き処理と同時処理を並行して行う画像デ  
ータ変換部を備えたことを特徴とするCMOSイメージセンサ。

【請求項5】

画素部がアレイ状に配置されてなるCMOSイメージセンサであって、  
センサの走査速度を出力画面上下のゆがみが気にならない速度にするとともに  
、該出力画面の最下部に対応するラインの走査終了から最上部に対応するライン

における走査開始までの間に、所定の走査停止期間を設けることにより、読み出しフレームを間引くように制御する読み出し駆動制御部を備えたことを特徴するCMOSイメージセンサ。

【請求項6】

画素部がアレイ状に配置されてなるCMOSイメージセンサであって、着目画素と縦方向及び横方向に隣接する4つの同色画素の画素データの平均値と該着目画素の画素データとの差分値が所定の不感帶範囲外に含まれる場合、該差分値に基づいて画像データの輪郭強調処理を行うことにより、該輪郭強調処理を行う時にノイズの強調を抑制する輪郭強調処理部を備えたことを特徴とするCMOSイメージセンサ。

【請求項7】

画素部がアレイ状に配置されてなるCMOSイメージセンサであって、フリッカレスになる積分時間として複数の値を設定し、該複数の値において積分時間が階段状に変化するように制御するとともに、積分時間が該複数の値の1つに固定されている場合には、アンプゲインが増加させるように制御する明るさ制御部を備えたことを特徴とするCMOSイメージセンサ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、撮像された画像の画質を向上させたCMOSイメージセンサに関するものである。

【0002】

【従来の技術】

(1) 第1の従来技術について説明する。

【0003】

図15は、従来のCMOSイメージセンサの画素部及びCDS回路の要部を示す図である。図15に示す回路の動作の詳細については、例えば特願2002-30613号の明細書に記載されている。

【0004】

従来のCMOSイメージセンサにおいては、アレイ状に配置される画素部のフォトダイオードにおいて光電変換された信号をアナログ信号として読み出し、読み出したアナログ信号をCDS回路（相関二重サンプリング回路）に入力する。また、CDS回路には、フォトダイオードのカソード側の電位をリセットするためのリセット動作におけるノイズ成分を示すリセットノイズ信号が入力される。

## 【0005】

(2) 第2の従来技術について説明する。

## 【0006】

図16は、従来のCMOSイメージセンサの欠陥補正回路の要部を示す図である。

## 【0007】

従来の欠陥補正回路においては、欠陥箇所格納用のROMを内蔵し、出荷試験時に欠陥と判定されたピクセルの位置情報を当該ROMに書き込んでおく。そして、当該ROMに書き込まれた位置情報に基づいて、欠陥ピクセル部分については、その周囲のピクセルの情報を用いて補間処理を行うことで補正していた。

## 【0008】

(3) 第3の従来技術について説明する。

## 【0009】

CMOSイメージセンサの画素データはAD変換でディジタル化された後、RGBそれぞれの感度補正、色補間処理、色調整、ガンマ変換、出力フォーマット変換などが施される。これらの処理をカラープロセッサで行っている。

## 【0010】

画素部がアレイ状に配置されたセンサ内部では、出力画像における同一縦列ピクセルデータが同じ回路、同じ信号経路を通っている。つまり、縦方向、つまり同一コラム毎に出力回路が設けられており、同一コラムのピクセルデータは同一の出力回路からカラープロセッサに出力される。

## 【0011】

そのため、出力回路がコラム毎に一定の特性、ゲイン、オフセットのずれを有する場合、それらのずれに起因して、CMOSイメージセンサの出力画像に縦方

向に縞模様が見えてしまう。このことはセンサ開発遅延の原因になる。

【0012】

(4) 第4の従来技術について説明する。

【0013】

図18は従来の1／3解像度変換の概要を説明するための図である。図19は従来の2／3解像度変換の概要を説明するための図である。

【0014】

従来、CMOSイメージセンサの出力画像から縮小画像を作る時的方法としては、イメージセンサ部で間引き処理を行い、後段のデジタル処理部にて同時処理を行う方法や、デジタル処理部で間引き処理及び同時処理を行うが、間引き処理と同時処理は別回路で行い時系列で処理する方法とがあった。

【0015】

例えば、1／3解像度変換の場合は、図18に示すように、 $6 \times 6$ 画素を1ブロックとして考え、各ブロック内で間引き処理を行って4画素分のデータを生成し、その後隣接するブロックの間引き処理後の4画素分のデータを用いて、間引き処理後の4画素分に対して同時処理を行う。

【0016】

尚、2／3解像度変換の場合の処理は図19に示したとおりであり、1／3解像度変換の場合と同様である。

【0017】

(5) 第5の従来技術について説明する。

【0018】

図20、21は、従来のCMOSイメージセンサにおける画像読み出し方式を説明するための図である。

【0019】

画素部がアレイ状に配置されたCMOSイメージセンサにおいては、従来、その画素読み出し方法としてローリングシャッターと呼ばれる、1ライン毎順々に読み出す駆動方式（電子シャッター）をとっている。

【0020】

図20、21は、上記読み出し駆動方式におけるセンサ上の走査ライン位置の時間変化を示している。尚、図21の場合の1フレーム分の時間は図20の場合のそれに比べて2倍になっている。

(6) 第6の従来技術について説明する。

#### 【0021】

従来、画像の輪郭を強調させるために、すべての画像データを対象にして、隣接する画素からの画像データ信号のレベル差を求め、この差を一律に予め決められた割合で増幅させることにより、隣接する画素の画像データ信号のレベル差を広げる処理を行っていた。

(7) 第7の従来技術について説明する。

#### 【0022】

室内の蛍光灯の発光周期等に起因して生じるフリッカノイズの低減方法としては、積分時間を蛍光灯の発光周期に応じて特定の値に変更させることで除去する方法が本願出願人により以前に特許出願されている。（特願2001-130150号参照。）

図22は、従来のオートゲインコントロール用デコーダにおける積分時間及びアンプゲインの制御方法を説明するための図である。尚、図中、右側に示したグラフは左側に示したグラフに対して横軸のスケールを拡大して表示したものである。

#### 【0023】

図22に示すように、従来は、光強度の比較的弱い室内での撮影では、積分時間を固定して、アンプゲインのみでオートゲインコントロールをかけることにより、被写体に対する明るさの制御を十分行うことができた。

#### 【0024】

#### 【発明の解決しようとする課題】

(1) 第1の従来技術の問題点を説明する。

#### 【0025】

図15において、アナログ信号とリセットノイズ信号のうちリセットノイズ信号においては、例えば太陽のような強い光が入ると、リセットノイズを読み出す

短い時間内でもリセットノイズ信号が積分されてしまう。その結果、読み出されたアナログ信号において本来の信号振幅がつぶされしまい、明るいはずの太陽が暗く写ってしまうといった問題があった。

## 【0026】

(2) 第2の従来技術の問題点を説明する。

## 【0027】

欠陥はある確率（ウェハプロセスの欠陥の発生率等）で発生するが、ROMを使用する場合、ROMの容量に限度があるため、想定以上の欠陥が発生すると、補正を行うことができないという問題があった。更に、経時変化によって出荷試験後に欠陥が増加するような場合には補正を行うことができないという問題があった。

## 【0028】

(3) 第3の従来技術の問題点を説明する。

## 【0029】

上述のカラープロセッサの処理のうち、RGB感度補正機能を拡張し、上記のコラム毎に現れるオフセット値を調節できる機能を持たせ、縦縞をキャンセルさせることは可能である。従来、図17に示すように、RGB感度補正是原色フィルターの場合、R、Gr、B、Gbの4つそれぞれ別個にゲインとオフセットを持たせることができ、これを実現するのに4系統の調整用レジスタを持たせていた。

## 【0030】

しかしながら、この4系統のレジスタはセンサの2コラム分に対応するのみであるため、例えば4コラムパターンで縦縞が発生する危険性がある場合には十分に対応することができず、縦縞模様のキャンセル効果が不充分であるという問題があった。

## 【0031】

(4) 第4の従来技術の問題点を説明する。

## 【0032】

しかしながら、デジタルズーム等の機能をカバーするには、縮小画面サイズも多様化になり、間引き方法も複雑にしなければいけなくなった。また、間引き処

理において画像情報に抜けがないようにするには、5ラインのラインメモリを必要とし、その後に行う同時処理においてもラインメモリを新たに2本必要とする。加えて、同時処理後の信号は、周辺 $12 \times 12$ 画素の情報が混じり合い、RGBそれぞれが位置情報の重みが変わってしまう。

#### 【0033】

以上の結果、間引き後に同時処理を行うと解像度が悪くなる、回路規模やラインメモリを多数必要とするといったような問題があった。

#### 【0034】

(5) 第5の従来技術の問題点を説明する。

#### 【0035】

ローリングシャッター方式では、図20から明らかなように、画面の最上部と最下部とでは、対応する露光時間（積分時間）にほぼ1フレーム分の時間のずれを生じていることになる。このことが原因となり、特に1フレームの時間が長い場合、撮像対象の物体（図20、21ではアルファベット文字のA）が高速に移動している状況では、図21に示したように、その物体に対する出力画像がゆがんでしまうという問題があった。

#### 【0036】

一方で1フレーム分の時間はCMOSイメージセンサのアプリケーションに応じて制約を受けるものであり、要求される積分時間の下限値なども異なってくるため、単純に1フレーム分の時間を短くすることができない場合がある。

#### 【0037】

このため、一定時間以上の積分時間を確保しながら、ゆがみのない画像を得ることが可能なCMOSイメージセンサが要求されている。

#### 【0038】

(6) 第6の従来技術の問題点を説明する。

#### 【0039】

すべての画像データを処理の対象とすると、画像のノイズも強調することになってしまふため、画質を劣化させる原因となるという問題があった。このため、画質を劣化させずに、所望の強調した画像を生成することが容易な輪郭強調処理

方法が要求されている。

【0040】

(7) 第7の従来技術の問題点を説明する。

【0041】

近年、CMOCイメージセンサの感度が向上したため、光強度の比較的弱い室内等の撮影でも、積分時間を変動させなければ明るさ制御が困難になっている。そのため、明るい室内で撮影を行うと、単純な積分時間制御では、フリッカノイズが発生する問題が発生した。

【0042】

詳しくは、従来方法では、ゲインが0dBになった後は、積分時間がリニアに減少させることで、被写体の明るさを一定に保つように調整する必要がある。しかしながら、積分時間がリニアに変化するとフリッカノイズが発生してしまう。

【0043】

本発明は上述の各問題点を鑑みてなされたものであり、撮像された画像の画質を向上させたCMOSイメージセンサを提供することを目的とする。

【0044】

【発明を解決するための手段】

(1) 第1の従来技術の問題点は、フォトダイオードと、少なくともリセットトランジスタ、增幅トランジスタ、水平方向選択トランジスタを備えたCMOSイメージセンサにおいて、リセットノイズの読み出し信号を監視し、任意の判定電圧に応じて信号をリセット電位に固定する検出回路を持つように構成することによって解決される。ここで、前記検出回路は、NOR回路することができる。検出回路はコンパレータで構成することができる。また、前記判定電圧を任意に可変可能に構成することができる。

【0045】

(2) 第2の従来技術の問題点は、CMOSイメージセンサにおいて以下のような動的に欠陥を補正する方法によって解決される。すなわち、動的に欠陥を補正する場合、欠陥でない画素に影響を与える可能性があるが、演算方法を工夫する事により欠陥のみを選択的に補正する方法を提供する。

## 【0046】

具体的には、上下左右の同色のピクセルとの演算を行う。また、不感帯を設けることにより正常なピクセルを欠陥補正対象から除外し、真の欠陥のみを補正する。不感帯の設定方法については、オフセットまたは、着目ピクセルのコードに比例した値、あるいはその組み合わせを使用し、それによって真の欠陥を見つけて出し補正する。補正する場合には、周囲との重み付けを変更可能にして、欠陥画素の色調を調整することも可能とする。

## 【0047】

(3) 第3の従来技術の問題点は、デジタルカラー処理回路において、カラーフィルタの色成分よりも多い数のレジスタを有し、それぞれのレジスタによりゲインとオフセットを調整可能に構成することにより解決される。

## 【0048】

更に、アナログアンプゲインに連動して上記オフセット値を調節可能に構成することができる。また、コラム毎に累積値を算出し、そのオフセット値を打ち消す機能をもつ回路を備えるようにすることも可能である。

## 【0049】

(4) 第4の従来技術の問題点は、CMOSイメージセンサにおいて、画像縮小後の仮想画素を想定して、その仮想画素を目標にして間引き処理と同時処理を行うことにより解決される。その際の計算は、各色の位置情報ができるだけ同じになるような重み付けを行うようとする。

## 【0050】

(5) 第5の従来技術の問題点は、CMOSイメージセンサにおいて、画面上下のゆがみが気にならない速度で走査し、読み出しフレームを間引くことで、積分時間を確保するように構成することにより解決される。

## 【0051】

更に、CMOSイメージセンサにおいて、ラインバッファを用いることで読み出し期間を引き延ばして、1フレームの間に均等に読み出し動作をするように構成することも可能である。

## 【0052】

(6) 第6の従来技術の問題点は、画像の輪郭強調処理方法において、該輪郭強調処理を行う時に不感帯を設けて強調したくない画像部分を処理対象外とすることによってノイズの強調を抑制するように構成することにより解決される。

#### 【0053】

(7) 第7の従来技術の問題点は、CMOSイメージセンサにおいて、フレームレートによってフリッカレスになる積分時間は数種類存在することを利用して、その積分時間とアンプゲインとを連動させたオートゲインコントロールを行うことにより解決される。これにより、フリッカレスとなる室内照度を3倍以上向上させることができた。

#### 【0054】

##### 【発明の実施の形態】

以下、図面を参照して本発明の実施の形態について説明する。しかしながら、係る実施の形態が本発明の技術的範囲を限定するものではなく、本発明の技術的範囲は特許請求の範囲とその均等物に及ぶものである。

#### 【0055】

(1) 本発明の第1の実施の形態について説明する。

#### 【0056】

図1は本発明の第1の実施の形態におけるCMOSイメージセンサの画素部及びCDS回路の要部を示す図である。図1においては、リセットノイズ信号を読みだしている期間、その信号レベルを監視し所定の判定電圧を超えて積分された場合に限り、そのリセットノイズ信号のレベルをリセット電位に固定する検出回路2を附加している。

#### 【0057】

図1に示すように、検出回路2はイネーブル信号がHレベルのときには、ノードN1の電位によらず、スイッチ回路SW4にHレベルを供給してオン状態にし、SW5にLレベルを供給してオフ状態に制御する。

#### 【0058】

一方、検出回路2は、イネーブル信号がLレベルのときにはノードN1の電位に応じて、ノードN1の電位がNOR回路の閾値よりも大きいときにはスイッチ

回路SW4にHレベルを供給してオン状態にし、SW5にLレベルを供給してオフ状態に制御するとともに、ノードN1の電位がNOR回路の閾値よりも小さいときにはスイッチ回路SW4にLレベルを供給してオフ状態にし、SW5にHレベルを供給してオン状態に制御し、それによってノードN1の電位をリセット電位VRに固定する。

## 【0059】

従って、本発明の第1の実施形態の回路を用いれば、強度の大きい光が入射された場合であっても、その強い光によってリセットノイズの信号が積分されず、リセット信号のレベルをリセット電圧に固定されるので、本来必要である光電変換された信号を取り出すことができ、強烈に明るい光を画像表現することが可能になる。

## 【0060】

(2) 本発明の第2の実施の形態について説明する。

## 【0061】

図2は本発明の第2の実施の形態における欠陥補正回路の要部を示す図である。図3はCMOSイメージセンサのピクセル配置を示す図である。図4は欠陥画素補正の計算に使用するピクセルを示す図である。

## 【0062】

以下に、欠陥補正回路4及び判定回路5によって行われる欠陥画素補正の計算について詳細に説明する。

## 【0063】

イメージセンサの出力信号がRGBベイヤ配列(図3)である場合には、計算に使用するピクセルには、図4のように着目画素 $p[x,y]$ と同色の縦横4画素を使用する。

## 【0064】

まず、着目ピクセル $p[x,y]$ のコードを周辺横方向2ピクセルと比較する。すなわち、以下の式1を満足するか否かを判定する。

## 【0065】

$$p[x-2,y] \leqq p[x,y] \leqq p[x+2,y], \text{かつ } p[x-2,y] \geqq p[x,y] \geqq p[x+2,y]$$

次に、式1を満足する時、正常ピクセルと判断する。すなわち、 $p[x,y] = p[x,y]$ とする。一方、式1を満足しない時、 $p[x,y]$ を横方向の（平均値土不感帯）と比較する。すなわち、以下の式2、式3を満足するか否かを判定する。（上から式2、3とする。）

$$p[x,y] < Wlimit \times (p[x-2,y] + p[x+2,y]) / 2 + Woffset$$

$$p[x,y] > Blimit \times (p[x-2,y] + p[x+2,y]) / 2 - Boffset$$

ここで、式2、式3を満足する時、正常ピクセルと判断する。すなわち、 $p[x,y] = p[x,y]$ とする。一方、式2、式3を満足しない時、 $p[x,y]$ を縦方向2ピクセルと比較する。すなわち、以下の式4を満足するか否かを判定する。

#### 【0066】

$$p[x,y-2] \leq p[x,y] \leq p[2,y+2] \text{、かつ } p[x,y-2] \geq p[x,y] \geq p[x,y+2]$$

ここで、式4を満足する時、正常ピクセルと判断する。すなわち、 $p[x,y] = p[x,y]$ とする。一方、式4を満足しない時、 $p[x,y]$ を縦方向の（平均値土不感帯）と比較する。すなわち、以下の式5、式6を満足するか否かを判定する。（上から式5、6とする。）

$$p[x,y] < Wlimit \times (p[x,y-2] + p[x,y+2]) / 2 + Woffset$$

$$p[x,y] > Blimit \times (p[x,y-2] + p[x,y+2]) / 2 - Boffset$$

ここで、式5、式6を満足する時、正常ピクセルと判断する。すなわち、 $p[x,y] = p[x,y]$ とする。一方、式5を満足しない時、欠陥画素（白点）と判断し、 $p[x,y]$ を以下の式7に従って補正する。

#### 【0067】

$$p[x,y] = p[x,y] \times Wweight$$

$$+ (p[x-2,y] + p[x+2,y] + p[x,y-2] + p[2,y+2]) / 4 \times (1 - Wweight)$$

また、式6を満足しない時、欠陥画素（黒点）と判断し、 $p[x,y]$ を以下の式8に従って補正する。

#### 【0068】

$$p[x,y] = p[x,y] \times Bweight$$

$$+ (p[x-2,y] + p[x+2,y] + p[x,y-2] + p[2,y+2]) / 4 \times (1 - Bweight)$$

上述の式7、8において、Wlimit、Blimitは、着目ピクセルのコードの大きさ

に比例した不感帯を与える為の係数である。Woffset、Woffsetは、着目ピクセルに対し一定の大きさになるように不感帯を与える為の係数である。Wlimit、Blimitは、同一の値でもよいし、それぞれ別の値を設定してもよい。Woffset、Woffsetは、同一の値でもよいし、それぞれ別の値を設定してもよい。

#### 【0069】

また、Wweight、Bweightは、着目ピクセルを補正する時に、周囲4ピクセルとの重みを変更する係数であり、補正する際のピクセルのコードを調整することができる。Wweight、Bweightは、同一の値でもよいし、それぞれ別の値を設定してもよい。

#### 【0070】

上述の実施例では、横方向の比較、縦方向の比較の順に計算しているが、順序を逆にしても同じ結果が得られる。また、縦方向、横方向の比較を同時に行う事も可能であり、以下の論理とすることにより同じ結果が得られる。すなわち、

式2、式5を満足しない時、欠陥画素（白点）と判断してp[x,y]を補正し、  
式3、式6を満足しない時、欠陥画素（黒点）と判断してp[x,y]を補正する。

#### 【0071】

従って、本発明の第2の実施形態の回路を用いれば、補正の対象となるピクセルに制限がないので、出荷試験の不良率を下げることができるとともに、ROMを内蔵する必要がないので、チップ面積を削減することができる。更に、経時変化によって出荷試験後に欠陥が増加するような場合であっても補正が可能となり、真の欠陥ピクセルのみを補正できるため、画像を高画質化できる。

#### 【0072】

(3) 本発明の第3の実施の形態について説明する。

#### 【0073】

図5は、本発明の第3の実施の形態におけるカラープロセッサの要部を示す図である。図6は、カラーフィルターの色成分の配列と各色成分に対するゲイン及びオフセット値の関係を説明するための図である。

#### 【0074】

第3の実施の形態におけるカラープロセッサでは、例えば4コラムパターンご

とに縦縞模様が発生する可能性がある場合には、8系統のゲイン及びオフセット値調整用のレジスタを設け、図6に示すように、4コラムに対してそれぞれ独立にゲイン及びオフセット値を調整することができるよう構成する。これにより、4コラムに1列のパターンで周期的に出現する縦縞を補正することが可能になる。

#### 【0075】

図5のカラープロセッサにおいて、係数セレクタ8、9はそれぞれタイミング発生部7の制御のもとに、少なくともカラーフィルターの色成分よりも多い数である複数のレジスタの中から、処理対象のピクセルデータの属するコラムに対応するレジスタを選択し、オフセット調整部10及びゲイン調整部11に対応するオフセット値及びゲインを供給する。

#### 【0076】

従って、本発明の第3の実施形態の回路を用いれば、縦方向別に個別に画像データを微調整することができ、出力画像から縦縞模様を補正し、鮮明な画像を得ることができる。たとえセンサからの出力画像データに縦縞模様が発生していたとしても、それを補正し、カメラモジュール全体として縦縞模様のない画像出力を得ることができる。

#### 【0077】

(4) 本発明の第4の実施の形態について説明する。

#### 【0078】

図7～11は本発明の第4の実施の形態における解像度変換と同時処理を説明するための図である。図7～10は解像度変換と同時処理を両方とも行う場合の計算方法を示すものである。図11は解像度変換を伴わず、同時処理のみを行う場合の計算方法を示すものである。

#### 【0079】

例えば、1/3解像度変換の場合は、図7に示すように、内部に数字を記した丸印で示したものが仮想画素に相当し、各々の仮想画素を目標にして図7に示された変換式に従って解像度変換と同時処理を行う。これにより、ラインメモリは2本しか必要とせず、しかも3×3画素の情報で各色の位置情報を等しい解像力

の良い処理が実行できる。

#### 【0080】

尚、 $2/3$ 、 $1/2$ 及び $1/4$ 解像度変換の場合の処理はそれぞれ、図8～10に示したとおりであり、 $1/3$ 解像度変換の場合と同様である。

従って、本発明の第4の実施形態の方法を用いれば、センサの出力画像から縮小画像を生成する際の間引き処理及び同時処理において、ラインメモリを削減しながら解像度の向上を実現することが可能になる。

(5) 本発明の第5の実施の形態について説明する。

#### 【0081】

図11は本発明の第5の実施の形態における画素読み出し方法を説明するための図である。

#### 【0082】

第5の実施の形態の画素読み出し方法においては、図11に示すように、単純に1フレーム分の時間を長くする（読み出し動作のためのクロックを遅くする）のではなく、光積分開始時間のずれを詰めるタイミングで各ラインごとに読み出し駆動を行い、その代わりに画面の最下部に対応するラインの走査終了から最上部に対応するラインにおける走査開始までの間に、所定の走査停止期間（読み出しのためのライン走査を停止させる期間）を設けるようにしている。

#### 【0083】

従って、本発明の第5の実施形態の方法を用いれば、一定時間以上の積分時間を確保しつつ、1フレーム分の時間を短くして実質ゆがみが気にならない画像を得ることが可能になる。

(6) 本発明の第6の実施の形態について説明する。

#### 【0084】

図13は、本発明の第6の実施の形態における輪郭強調処理のための計算に使用する画像ピクセル位置を説明するための図である。

#### 【0085】

以下に第6の実施の形態の1例である平均方式（弱い強調）について説明する。この平均方式は、着目ピクセルのG[x,y]と周辺ピクセルGの平均値との差分を

強調することにより、輪郭強調処理を行うものである。

#### 【0086】

まず、周辺のGの平均値との差分を計算し、比較する。すなわち、以下の式1を満足するか否かを判定する。

#### 【0087】

$\text{lowlimit} < (G[x-1,y] + G[x+1,1] + G[x,y-1] + G[x,y+1]) / 4 - G[x,y] < \text{maxlimit}$

t

ここで、式1を満足する時は強調しない。すなわち、 $R[x,y] = R[x,y]$ 、 $G[x,y] = G[x,y]$ 、 $B[x,y] = B[x,y]$ とする。

#### 【0088】

一方、式1を満足しない時、次式により強調する。

#### 【0089】

$R[x,y] = R[x,y]$

$-R\text{gain} \times ((G[x-1,y] + G[x+1,1] + G[x,y-1] + G[x,y+1]) / 4 - G[x,y])$

$G[x,y] = G[x,y]$

$-G\text{gain} \times ((G[x-1,y] + G[x+1,1] + G[x,y-1] + G[x,y+1]) / 4 - G[x,y])$

$B[x,y] = B[x,y]$

$-B\text{gain} \times ((G[x-1,y] + G[x+1,1] + G[x,y-1] + G[x,y+1]) / 4 - G[x,y])$

ここで、 $\text{lowlimit}$ 及び $\text{maxlimit}$ は、着目ピクセルの $G[x,y]$ と周辺ピクセルGの平均値との差分の下限値及び上限値を与えるための係数である。 $R\text{gain}$ 、 $G\text{gain}$ 及び $B\text{gain}$ は各画素の色成分ごとの強調の割合を与えるための係数である。

#### 【0090】

以下に第6の実施の形態の1例である最大値抽出方式（強い強調）について説明する。この最大値抽出方式は着目ピクセルの $G[x,y]$ と周辺ピクセルGとの差分が最も大きい値を使用して強調することにより、輪郭強調処理を行うものである。

#### 【0091】

まず、周辺のGとの差分を計算し、比較する。すなわち、以下の式1～4を満足するか否かを判定する。（上から式1～4とする。）

$\text{lowlimit} < G[x-1, y] - G[x, y] < \text{maxlimit}$

$\text{lowlimit} < G[x+1, y] - G[x, y] < \text{maxlimit}$

$\text{lowlimit} < G[x, y-1] - G[x, y] < \text{maxlimit}$

$\text{lowlimit} < G[x, y+1] - G[x, y] < \text{maxlimit}$

ここで、式1～4をすべて満足する時は強調しない。すなわち、 $R[x, y] = R[x, y]$ 、 $G[x, y] = G[x, y]$ 、 $B[x, y] = B[x, y]$ とする。

#### 【0092】

一方、式1～4を少なくとも1つ満足しない時、式1～4の計算結果におけるを使用して次式で強調する。

#### 【0093】

$R[x, y] = R[x, y] - R\text{gain} \times [G \text{の差分の最大値(符号付き)}]$

$G[x, y] = G[x, y] - G\text{gain} \times [G \text{の差分の最大値(符号付き)}]$

$B[x, y] = B[x, y] - B\text{gain} \times [G \text{の差分の最大値(符号付き)}]$

また、式1～4の差分の計算結果において、着目ピクセル $G[x, y]$ が周辺ピクセル $G$ より小さい場合の「絶対値の最大値」と着目ピクセル $G[x, y]$ が周辺ピクセル $G$ より大きい場合の「絶対値の最大値」が同じ値になる組み合わせが存在する場合は強調しない。すなわち、 $R[x, y] = R[x, y]$ 、 $G[x, y] = G[x, y]$ 、 $B[x, y] = B[x, y]$ とする。尚、この場合は、式1～4を満足するか否かは考慮しない。

#### 【0094】

従って、本発明の第6の実施形態の方法を用いれば、不感帯を設けることできうるので、画像のノイズの強調を防止することができ、画像の高画質化が可能となる。

#### 【0095】

(7) 本発明の第7の実施の形態について説明する。

#### 【0096】

図14は、本発明の第7の実施の形態におけるオートゲインコントロール用デコーダにおける積分時間及びアンプゲインの制御方法を説明するための図である。尚、図中、右側に示したグラフは左側に示したグラフに対して横軸のスケールを拡大して表示したものである。図14は一例として30fps用のデコーダに関

するものである。

#### 【0097】

図14に示すように、フリッカレスになる積分時間として50Hz用では3種類、60Hz用では4種類の値を設定し、これらの値を利用して積分時間が階段状に変化するように制御されている。一方、図14に示すように、上述の積分時間の階段状変化を補うように、積分時間が一定値に固定されている区間においては、局所的にアンプゲインが0dBよりも大きな値に変化するように制御されている。

#### 【0098】

従って、本発明の第7の実施形態の方法を用いれば、滑らかでフリッカレスなオートゲインコントロールが明るい室内でも行うことができる。

#### 【0099】

##### 【発明の効果】

以上説明したように、本発明によれば、CMOSイメージセンサにおいて撮像された画像の画質を大きく向上させることができると効果を奏するものであり、係るCMOSイメージセンサの性能向上に寄与するところが極めて大きい。

##### 【図面の簡単な説明】

【図1】 本発明の第1の実施の形態におけるCMOSイメージセンサの画素部及びCDS回路の要部を示す図

【図2】 本発明の第2の実施の形態における欠陥補正回路の要部を示す図

【図3】 CMOSイメージセンサのピクセル配置を示す図

【図4】 欠陥画素補正の計算に使用するピクセルを示す図

【図5】 本発明の第3の実施の形態におけるカラープロセッサの要部を示す図

【図6】 カラーフィルターの色成分の配列と各色成分に対するゲイン及びオフセット値の関係を説明するための図

【図7】 本発明の第4の実施の形態における解像度変換と同時処理を説明するための図

【図8】 本発明の第4の実施の形態における解像度変換と同時処理を説明するための図

【図9】 本発明の第4の実施の形態における解像度変換と同時処理を説明するための図

【図10】 本発明の第4の実施の形態における解像度変換と同時処理を説明するための図

【図11】 本発明の第4の実施の形態における解像度変換と同時処理を説明するための図

【図12】 本発明の第5の実施の形態における画素読み出し方法を説明するための図

【図13】 本発明の第6の実施の形態における輪郭強調処理のための計算に使用する画像ピクセル位置を説明するための図

【図14】 本発明の第7の実施の形態におけるオートゲインコントロール用デコーダにおける積分時間及びアンプゲインの制御方法を説明するための図

【図15】 従来のCMOSイメージセンサの画素部及びCDS回路の要部を示す図

【図16】 従来のCMOSイメージセンサの欠陥補正回路の要部を示す図

【図17】 従来のカラーフィルターの色成分の配列と各色成分に対するゲイン及びオフセット値の関係を説明するための図

【図18】 従来の1/3解像度変換の概要を説明するための図

【図19】 従来の2/3解像度変換の概要を説明するための図

【図20】 従来のCMOSイメージセンサにおける画像読み出し方式

【図21】 従来のCMOSイメージセンサにおける画像読み出し方式

【図22】 従来のオートゲインコントロール用デコーダにおける積分時間及びアンプゲインの制御方法を説明するための図

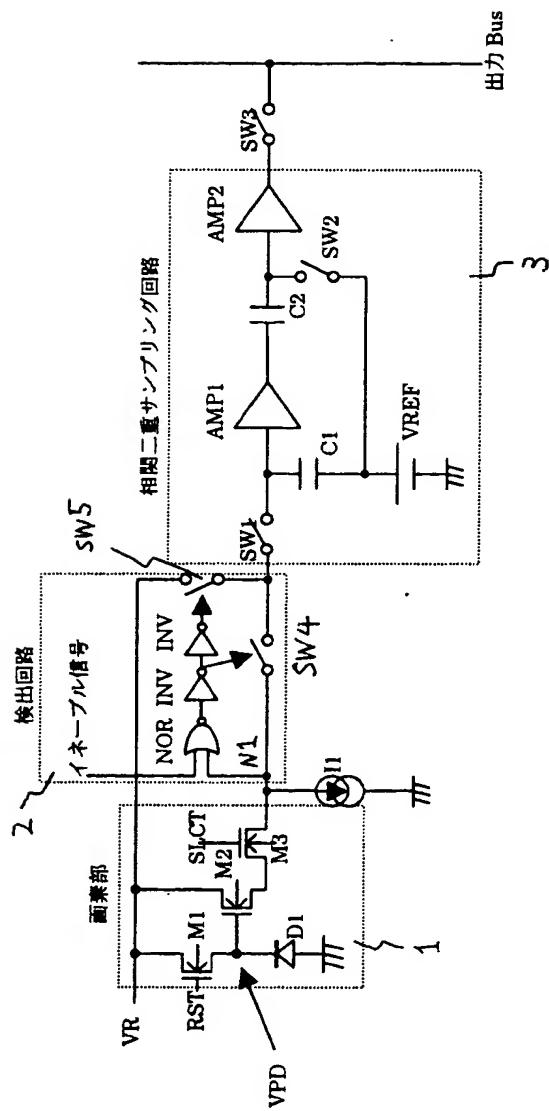
#### 【符号の説明】

- 1 画素部、
- 2 検出回路、
- 3 相関二重サンプリング回路、
- 4 欠陥補正回路、
- 5 判定回路、

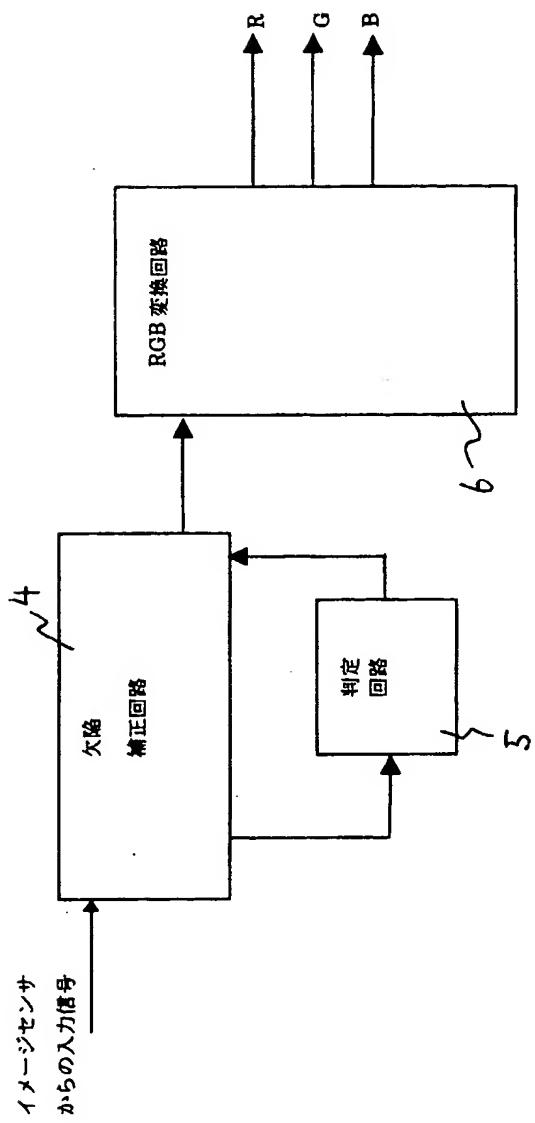
- 6 RGB 変換回路、
- 7 タイミング発生部、
- 8 係数セレクタ、
- 9 係数セレクタ、
- 10 オフセット調整部、
- 11 ゲイン調整部

【書類名】 図面

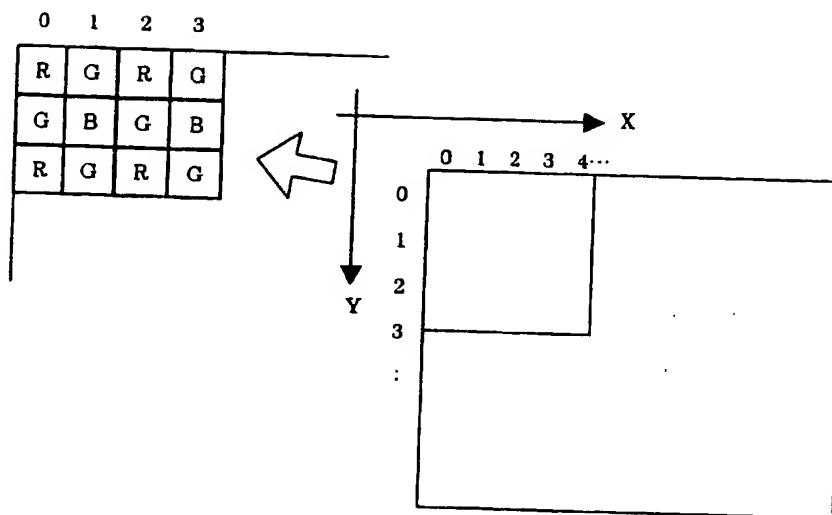
【図1】



【図2】



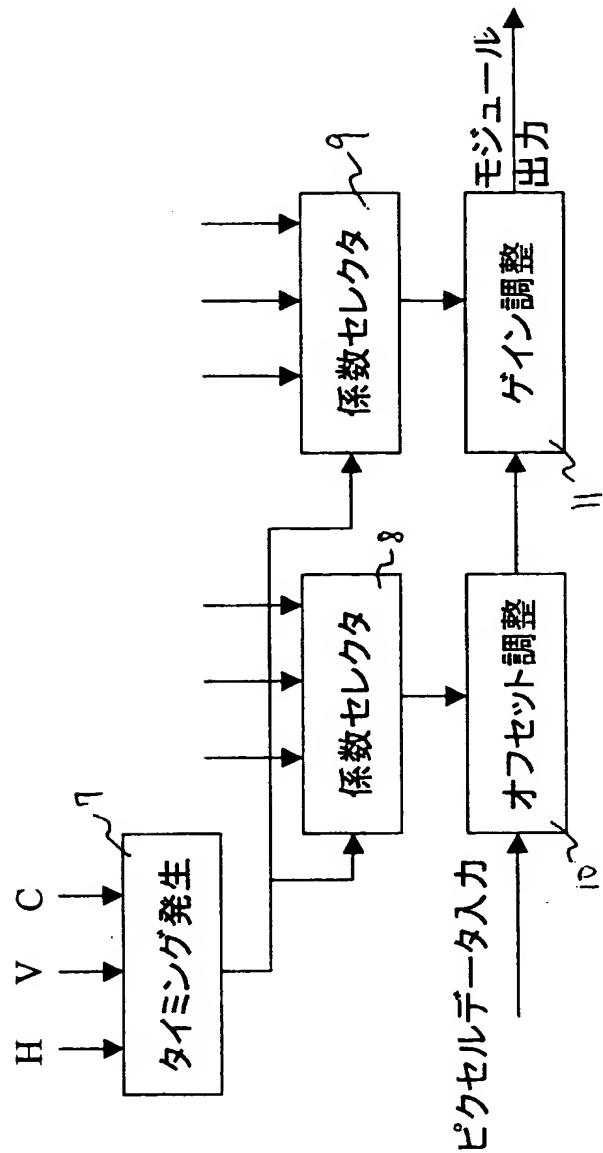
【図3】



【図4】

		$p[x,y-2]$		
$p[x-2,y]$		$p[x,y]$		$p[x+2,y]$
		$p[x,y+2]$		

【図5】



【図6】

R1 G1 R2 G2 R1 G1

G3 B1 G4 B2 G3 B1

R1 G1 R2 G2 R1 G1

G3 B1 G4 B2 G3 B1

R1のタイミングで Gain1 offset1

G1のタイミングで Gain2 offset2

R2のタイミングで Gain3 offset3

G2のタイミングで Gain4 offset4

G3のタイミングで Gain5 offset5

B1のタイミングで Gain6 offset6

G4のタイミングで Gain7 offset7

B2のタイミングで Gain8 offset8

【図7】

1/3変換

解像度変換後の信号が擬似ピクセル位置(図中○)になるように同時処理を行う。

RGB別の変換式は下記のとおりとなる。

計算に使用するピクセル

B0.0	G1.0	B2.0	G3.0	B4.0	G5.0	B6.0	G7.0	B8.0	G9.0	B10.0	G11.0	B12.0
G0.1	R1.0	G2.1	R3.1	G4.1	R5.1	G6.1	R7.1	G8.1	R9.1	G10.1	R11.1	G12.1
B0.2	G1.2	B2.2	G3.2	B4.2	G5.2	B6.2	G7.2	B8.2	G9.2	B10.2	G11.2	B12.2
G0.3	R1.3	G2.3	R3.3	G4.3	R5.3	G6.3	R7.3	G8.3	R9.3	G10.3	R11.3	G12.3
B0.4	R1.4	B2.4	G3.4	R4.4	G5.4	R6.4	G7.4	B8.4	G9.4	R10.4	G11.4	B12.4
G0.5	R1.5	G2.5	R3.5	G4.5	R5.5	G6.5	R7.5	G8.5	R9.5	G10.5	R11.5	G12.5
B0.6	G1.6	B2.6	G3.6	B4.6	G5.6	B6.6	G7.6	B8.6	G9.6	B10.6	G11.6	B12.6
G0.7	R1.7	G2.7	R3.7	G4.7	R5.7	G6.7	R7.7	G8.7	R9.7	G10.7	R11.7	G12.7
B0.8	G1.8	B2.8	G3.8	B4.8	G5.8	B6.8	G7.8	B8.8	G9.8	B10.8	G11.8	B12.8
G0.9	R1.9	G2.9	R3.9	G4.9	R5.9	G6.9	R7.9	G8.9	R9.9	G10.9	R11.9	G12.9
B0.10	R1.10	B2.10	G3.10	B4.10	G5.10	B6.10	G7.10	B8.10	G9.10	R10.10	G11.10	B12.10
G0.11	R1.11	G2.11	R3.11	G4.11	R5.11	G6.11	R7.11	G8.11	R9.11	G10.11	R11.11	G12.11
B0.12	G1.12	B2.12	G3.12	B4.12	G5.12	B6.12	G7.12	B8.12	G9.12	B10.12	G11.12	B12.12

①  $R'1.1 = (1^*R1.1)/1$   
 $G'1.1 = (1^*G1.0+1^*G0.1+1^*G2.1+1^*G1.2)/4$   
 $B'1.1 = (1^*B0.0+1^*B2.0+1^*B0.2+1^*B2.2)/4$

②  $R'2.1 = (1^*R3.1+1^*R5.1)/2$   
 $G'2.1 = (1^*G3.0+1^*G5.0+4^*G4.1+1^*G3.2+1^*G5.2)/8$   
 $B'2.1 = (1^*B4.0+1^*B4.2)/2$

③  $R'1.2 = (1^*R1.3+1^*R1.5)/2$   
 $G'1.2 = (1^*G0.3+1^*G2.3+4^*G1.4+1^*G0.5+1^*G2.5)/8$   
 $B'1.2 = (1^*B0.4+1^*B2.4)/2$

④  $R'2.2 = (1^*R3.3+1^*R5.3+1^*R3.5+1^*R5.6)/4$   
 $G'2.2 = (1^*G4.3+1^*G3.4+1^*G5.4+1^*G4.5)/4$   
 $B'2.2 = (1^*B4.4)/1$

\* 注)解像度変換後の2×2画素を1ブロックとして考えると、計算式はこの繰り返し計算になる。  
 2/3変換の単純間引きになる。

【図 8】

## 2/3変換

解 度変換後の信号が擬似ピクセル位置(図中○)になるように同時処理を行う。

RGB別の変換式は下記のとおりとなる。

## 計算に使用するピクセル

B0.0	G1.0	B2.0	G3.0	B4.0	G5.0	B6.0	G7.0	B8.0	G9.0	B10.0	G11.0	B12.0
G0.1	R1.0	G2.1	R3.1	G3.1	R5.1	G6.1	R6.1	G8.1	R9.1	G9.1	R11.0	B12.1
B0.2	G1.2	B2.2	G3.2	B4.2	G5.2	B6.2	G7.2	B8.2	G9.2	B10.2	G11.2	B12.2
G0.3	R1.3	G2.3	R3.3	G3.3	R5.3	G6.3	R7.3	G8.3	R9.3	G9.3	R11.3	B12.3
B0.4	G1.4	B2.4	G3.4	B4.4	G5.4	B6.4	G6.4	B8.4	G9.4	B10.4	G11.4	B12.4
G0.5	R1.5	G2.5	R3.5	G3.5	R5.5	G6.5	R7.5	G8.5	R9.5	G9.5	R11.5	B12.5
B0.6	G1.6	B2.6	G3.6	B4.6	G5.6	B6.6	G7.6	B8.6	G9.6	B10.6	G11.6	B12.6
G0.7	R1.7	G2.7	R3.7	G3.7	R5.7	G6.7	R6.7	G8.7	R9.7	G9.7	R11.7	B12.7
B0.8	G1.8	B2.8	G3.8	B4.8	G5.8	B6.8	G7.8	B8.8	G9.8	B10.8	G11.8	B12.8
G0.9	R1.9	G2.9	R3.9	G3.9	R5.9	G6.9	R7.9	G8.9	R9.9	G9.9	R11.9	B12.9
B0.10	G1.0	B2.10	G3.10	B4.10	G5.10	B6.10	G6.10	B8.10	G9.10	B10.10	G11.10	B12.10
G0.11	R1.11	G2.11	R3.11	G3.11	R5.11	G6.11	R6.11	G8.11	R9.11	G9.11	R11.11	B12.11
B0.12	G1.12	B2.12	G3.12	B4.12	G5.12	B6.12	G7.12	B8.12	G9.12	B10.12	G11.12	B12.12

- ①  $\begin{cases} R'1.1 = (1 * R1.1) / 4 \\ G'1.1 = (1 * G1.0 + 1 * G0.1 + 1 * G2.1 + 1 * G1.2) / 4 \\ B'1.1 = (1 * B0.0 + 1 * B2.0 + 1 * B0.2 + 1 * B2.2) / 4 \\ R'2.1 = (1 * R1.1 + 3 * R3.1) / 4 \end{cases}$
- ②  $\begin{cases} G'2.1 = (5 * G2.1 + 1 * G3.0 + 1 * G4.1 + 1 * G3.2) / 8 \\ B'2.1 = (3 * B2.0 + 1 * B4.0 + 3 * B2.2 + 1 * B4.2) / 8 \\ R'3.1 = (1 * R3.1 + 1 * R5.1) / 2 \end{cases}$
- ③  $\begin{cases} G'3.1 = (1 * G3.0 + 1 * G5.0 + 4 * G4.1 + 1 * G3.2 + 1 * G5.2) / 8 \\ B'3.1 = (1 * B4.0 + 1 * B4.2) / 2 \\ R'4.1 = (3 * R5.1 + 1 * R7.1) / 4 \end{cases}$
- ④  $\begin{cases} G'4.1 = (1 * G5.0 + 1 * G4.1 + 5 * G6.1 + 1 * G5.2) / 8 \\ B'4.1 = (1 * B4.0 + 3 * B6.0 + 1 * B4.2 + 3 * B6.2) / 8 \\ R'1.2 = (1 * R1.1 + 3 * R1.3) / 4 \end{cases}$
- ⑤  $\begin{cases} G'1.2 = (5 * G1.2 + 1 * G0.3 + 1 * G2.3 + 1 * G1.4) / 8 \\ B'1.2 = (3 * B0.2 + 3 * B2.2 + 1 * B0.4 + 1 * B2.4) / 8 \\ R'2.2 = (1 * R1.1 + 3 * R3.1 + 3 * R1.3 + 9 * R3.3) / 16 \end{cases}$
- ⑥  $\begin{cases} G'2.2 = (3 * G3.2 + 3 * G2.3 + 1 * G4.3 + 1 * G3.4) / 8 \\ B'2.2 = (3 * B2.2 + 3 * B4.2 + 3 * B2.4 + 1 * B4.4) / 16 \\ R'3.2 = (1 * R3.1 + 3 * R3.3 + 1 * R5.1 + 3 * R5.3) / 8 \end{cases}$
- ⑦  $\begin{cases} G'3.2 = (5 * G3.2 + 5 * G5.2 + 4 * G4.3 + 1 * G3.4 + 1 * G5.4) / 16 \\ B'3.2 = (3 * B4.2 + 1 * B4.4) / 4 \\ R'4.2 = (3 * R5.1 + 9 * R6.3 + 1 * R7.1 + 3 * R7.3) / 16 \end{cases}$
- ⑧  $\begin{cases} G'4.2 = (3 * G5.2 + 1 * G4.3 + 3 * G6.3 + 1 * G5.4) / 8 \\ B'4.2 = (3 * B4.2 + 9 * B6.2 + 1 * B4.4 + 3 * B6.4) / 16 \\ R'1.3 = (1 * R1.3 + 1 * R1.5) / 2 \end{cases}$
- ⑨  $\begin{cases} G'1.3 = (1 * G0.3 + 1 * G2.3 + 4 * G1.4 + 1 * G0.5 + 1 * G2.5) / 8 \\ B'1.3 = (1 * B0.4 + 1 * B2.4) / 2 \\ R'2.3 = (1 * R1.3 + 3 * R3.3 + 1 * R1.5 + 3 * R3.5) / 8 \end{cases}$
- ⑩  $\begin{cases} G'2.3 = (5 * G2.3 + 5 * G2.5 + 4 * G3.4 + 1 * G4.3 + 1 * G4.5) / 16 \\ B'2.3 = (3 * B2.4 + 1 * B4.4) / 4 \\ R'3.3 = (1 * R3.3 + 1 * R5.3 + 1 * R3.5 + 1 * R5.5) / 4 \end{cases}$
- ⑪  $\begin{cases} G'3.3 = (1 * G4.3 + 1 * G3.4 + 1 * G5.4 + 1 * G4.5) / 4 \\ B'3.3 = (1 * B4.4) / 1 \\ R'4.3 = (3 * R5.3 + 3 * R5.5 + 1 * R7.3 + 1 * R7.5) / 8 \end{cases}$
- ⑫  $\begin{cases} G'4.3 = (5 * G6.3 + 5 * G6.5 + 4 * G5.4 + 1 * G4.3 + 1 * G4.5) / 16 \\ B'4.3 = (1 * B4.4 + 3 * B6.4) / 4 \\ R'1.4 = (3 * R1.5 + 1 * R1.7) / 4 \end{cases}$
- ⑬  $\begin{cases} G'1.4 = (1 * G1.4 + 1 * G0.5 + 5 * G1.6 + 1 * G2.5) / 8 \\ B'1.4 = (1 * B0.4 + 1 * B2.4 + 3 * B0.6 + 3 * B2.6) / 8 \\ R'2.4 = (3 * R1.5 + 9 * R3.5 + 1 * R1.7 + 3 * R3.7) / 16 \end{cases}$
- ⑭  $\begin{cases} G'2.4 = (1 * G3.4 + 3 * G2.5 + 1 * G4.5 + 3 * G3.6) / 8 \\ B'2.4 = (3 * B2.4 + 1 * B4.4 + 9 * B2.6 + 3 * B4.6) / 16 \\ R'3.4 = (3 * R3.5 + 3 * R5.5 + 1 * R3.7 + 1 * R5.7) / 8 \end{cases}$
- ⑮  $\begin{cases} G'3.4 = (5 * G3.6 + 5 * G5.6 + 4 * G4.5 + 1 * G3.4 + 1 * G5.4) / 16 \\ B'3.4 = (1 * B4.4 + 3 * B4.6) / 4 \\ R'4.4 = (9 * R5.5 + 3 * R7.5 + 3 * R5.7 + 1 * R7.7) / 16 \end{cases}$
- ⑯  $\begin{cases} G'4.4 = (3 * G6.5 + 3 * G6.6 + 1 * G4.5 + 1 * G5.4) / 8 \\ B'4.4 = (1 * B4.4 + 3 * B6.4 + 3 * B4.6 + 9 * B6.6) / 16 \end{cases}$

\* 注)解像度変換後の4×4画素を1ブロックとして考えると、計算式はこの繰り返し計算になる。

【図9】

## 1/2変換

解像度変換後の信号が擬似ピクセル位置(図中○)になるように同時処理を行う。

RGB別の変換式は下記のとおりとなる。

## 計算に使用するピクセル

B0.0	G1.0	B2.0	G3.0	B4.0	G5.0	B6.0	G7.0	B8.0	G9.0	B10.0	G11.0	B12.0
G0.1	R1.1	G2.1	R3.1	G4.1	R5.1	G6.1	R7.1	G8.1	R9.1	G10.1	R11.1	G12.1
B0.2	G1.2	B2.2	G3.2	B4.2	G5.2	B6.2	G7.2	B8.2	G9.2	B10.2	G11.2	B12.2
G0.3	R1.3	G2.3	R3.3	G4.3	R5.3	G6.3	R7.3	G8.3	R9.3	G10.3	R11.3	G12.3
B0.4	G1.4	B2.4	G3.4	B4.4	G5.4	B6.4	G7.4	B8.4	G9.4	B10.4	G11.4	B12.4
G0.5	R1.5	G2.5	R3.5	G4.5	R5.5	G6.5	R7.5	G8.5	R9.5	G10.5	R11.5	G12.5
B0.6	G1.6	B2.6	G3.6	B4.6	G5.6	B6.6	G7.6	B8.6	G9.6	B10.6	G11.6	B12.6
G0.7	R1.7	G2.7	R3.7	G4.7	R5.7	G6.7	R7.7	G8.7	R9.7	G10.7	R11.7	G12.7
B0.8	G1.8	B2.8	G3.8	B4.8	G5.8	B6.8	G7.8	B8.8	G9.8	B10.8	G11.8	B12.8
G0.9	R1.9	G2.9	R3.9	G4.9	R5.9	G6.9	R7.9	G8.9	R9.9	G10.9	R11.9	G12.9
B0.10	G1.10	B2.10	G3.10	B4.10	G5.10	B6.10	G7.10	B8.10	G9.10	B10.10	G11.10	B12.10
G0.11	R1.11	G2.11	R3.11	G4.11	R5.11	G6.11	R7.11	G8.11	R9.11	G10.11	R11.11	G12.11
B0.12	G1.12	B2.12	G3.12	B4.12	G5.12	B6.12	G7.12	B8.12	G9.12	B10.12	G11.12	B12.12

$$\begin{cases} R'1.1 = (9*R1.1 + 3*R3.1 + 3*R1.3 + 1*R3.3)/16 \\ G'1.1 = (3*G2.1 + 3*G1.2 + 1*G3.2 + 1*G2.3)/8 \\ B'1.1 = (1*B0.0 + 3*B0.2 + 3*B2.0 + 9*B2.2)/16 \end{cases}$$

\* 注) 全画素この計算式になる。

【図10】

## 1/4変換

解像度変換後の信号が擬似ピクセル位置(図中○)になるように同時処理を行う。

RGB別の変換式は下記のとおりとなる。

## 計算に使用するピクセル

B0.0	G1.0	B2.0	G3.0	B4.0	G5.0	B6.0	G7.0	B8.0	G9.0	B10.0	G11.0	B12.0
G0.1	R1.1	G2.1	R3.1	G4.1	R5.1	G6.1	R7.1	G8.1	R9.1	G10.1	R11.1	G12.1
B0.2	G1.2	B2.2	G3.2	B4.2	G5.2	B6.2	G7.2	B8.2	G9.2	B10.2	G11.2	B12.2
G0.3	R1.3	G2.3	R3.3	G4.3	R5.3	G6.3	R7.3	G8.3	R9.3	G10.3	R11.3	G12.3
B0.4	G1.4	B2.4	G3.4	B4.4	G5.4	B6.4	G7.4	B8.4	G9.4	B10.4	G11.4	B12.4
G0.5	R1.5	G2.5	R3.5	G4.5	R5.5	G6.5	R7.5	G8.5	R9.5	G10.5	R11.5	G12.5
B0.6	G1.6	B2.6	G3.6	B4.6	G5.6	B6.6	G7.6	B8.6	G9.6	B10.6	G11.6	B12.6
G0.7	R1.7	G2.7	R3.7	G4.7	R5.7	G6.7	R7.7	G8.7	R9.7	G10.7	R11.7	G12.7
B0.8	G1.8	B2.8	G3.8	B4.8	G5.8	B6.8	G7.8	B8.8	G9.8	B10.8	G11.8	B12.8
G0.9	R1.9	G2.9	R3.9	G4.9	R5.9	G6.9	R7.9	G8.9	R9.9	G10.9	R11.9	G12.9
B0.10	G1.10	B2.10	G3.10	B4.10	G5.10	B6.10	G7.10	B8.10	G9.10	B10.10	G11.10	B12.10
G0.11	R1.11	G2.11	R3.11	G4.11	R5.11	G6.11	R7.11	G8.11	R9.11	G10.11	R11.11	G12.11
B0.12	G1.12	B2.12	G3.12	B4.12	G5.12	B6.12	G7.12	B8.12	G9.12	B10.12	G11.12	B12.12

$$\begin{cases} R'1.1 = (9*R1.1 + 3*R3.1 + 3*R1.3 + 1*R3.3)/16 \\ G'1.1 = (3*G2.1 + 3*G1.2 + 1*G3.2 + 1*G2.3)/8 \\ B'1.1 = (1*B0.0 + 3*B0.2 + 3*B2.0 + 9*B2.2)/16 \end{cases}$$

\* 注) 全画素この計算式になります。1/2変換の単純間引きになる。

【図11】

解像度変換なし。

解像度変換を行わないので、各ピクセルにRGB同時信号処理を行う。補間式は下記のとおりとする。

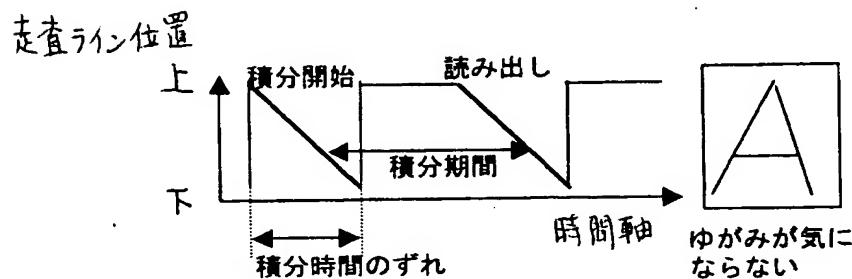
計算に使用するピクセル

B0.0	G1.0	B2.0	G3.0	B4.0	G5.0	B6.0
G0.1	R1.1	G2.1	R3.1	G4.1	R5.1	G6.1
B0.2	G1.2	B2.2	G3.2	B4.2	G5.2	B6.2
G0.3	R1.3	G2.3	R3.3	G4.3	R5.3	G6.3
B0.4	G1.4	B2.4	G3.4	B4.4	G5.4	B6.4
G0.5	R1.5	G2.5	R3.5	G4.5	R5.5	G6.5
B0.6	G1.6	B2.6	G3.6	B4.6	G5.6	B6.6

- ①. 1  $\begin{cases} R'1.1 = (1 * R1.1) / 1 \\ G'1.1 = (1 * G1.0 + 1 * G0.1 + 1 * G2.1 + 1 * G1.2) / 4 \\ B'1.1 = (1 * B0.0 + 1 * B2.0 + 1 * B0.2 + 1 * B2.2) / 4 \end{cases}$   
 $R'2.1 = (1 * R1.1 + 1 * R3.1) / 2$   
 $G'2.1 = (1 * G1.0 + 1 * G3.0 + 4 * G2.0 + 1 * G1.2 + 1 * G3.2) / 8$   
 $B'2.1 = (1 * B2.0 + 1 * B2.2) / 2$   
 $R'1.2 = (1 * R1.1 + 1 * R1.3) / 2$   
 $G'1.2 = (1 * G0.1 + 1 * G2.1 + 4 * G1.2 + 1 * G0.3 + 1 * G2.3) / 8$   
 $B'1.2 = (1 * B0.2 + 1 * B2.2) / 2$   
 $R'2.2 = (1 * R1.1 + 1 * R3.1 + 1 * R1.3 + 1 * R3.3) / 4$   
 $G'2.2 = (1 * G1.0 + 1 * G3.0 + 1 * G1.2 + 1 * G3.2) / 4$   
 $B'2.2 = (1 * B2.2) / 1$

\* 注) 2×2画素を1ブロックとして考えると、計算式はこの繰り返し計算になる。

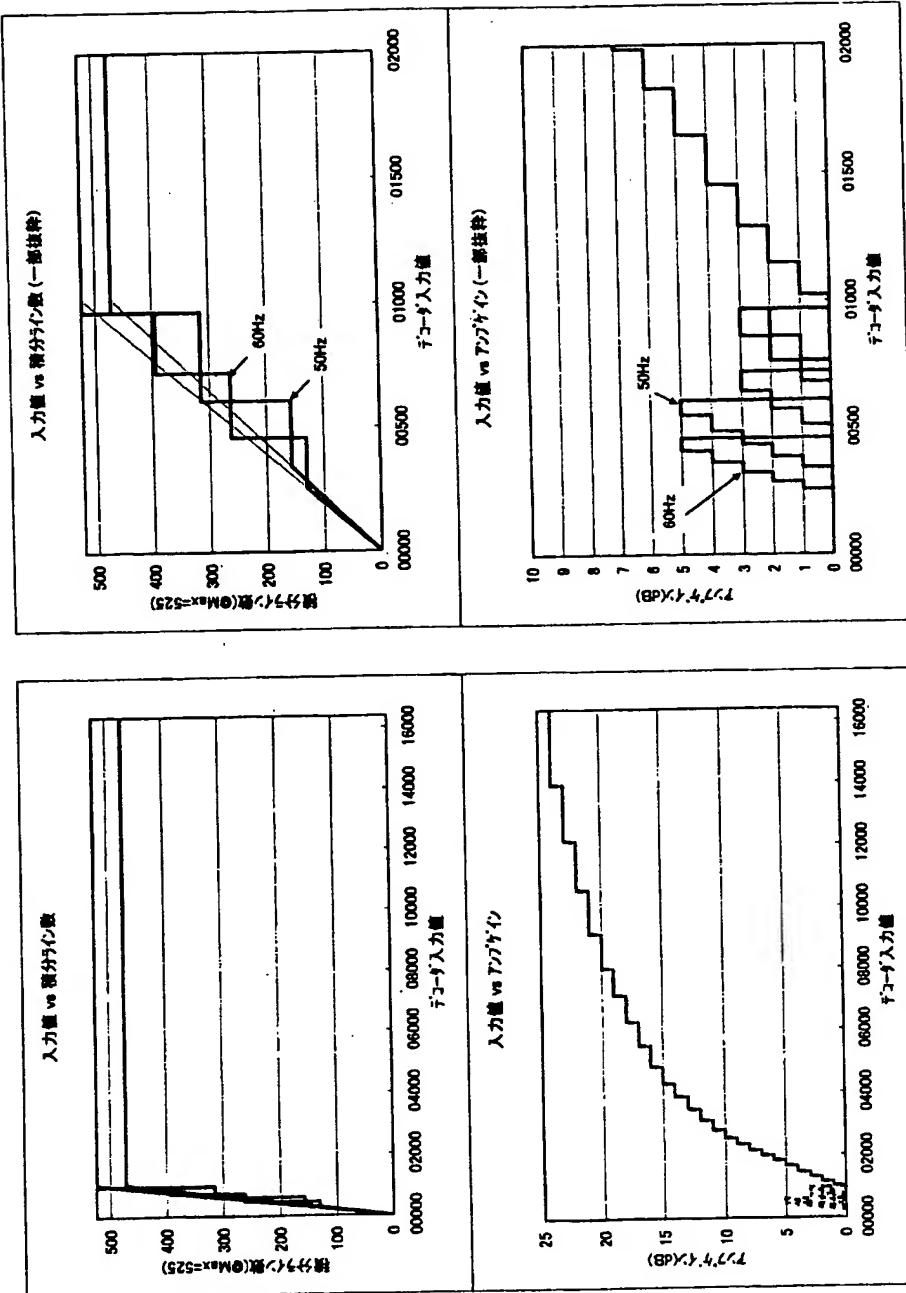
【図12】



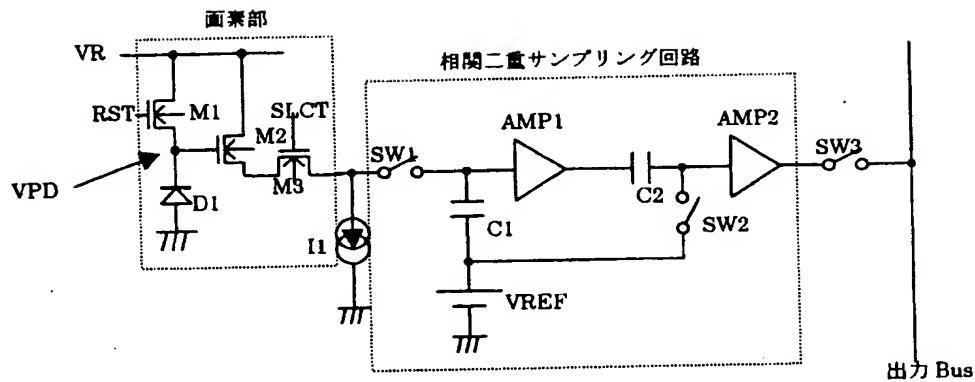
【図13】

	$p[x, y-1]$	
$p[x-1, y]$	$p[x, y]$	$p[x+1, y]$
	$p[x, y+1]$	

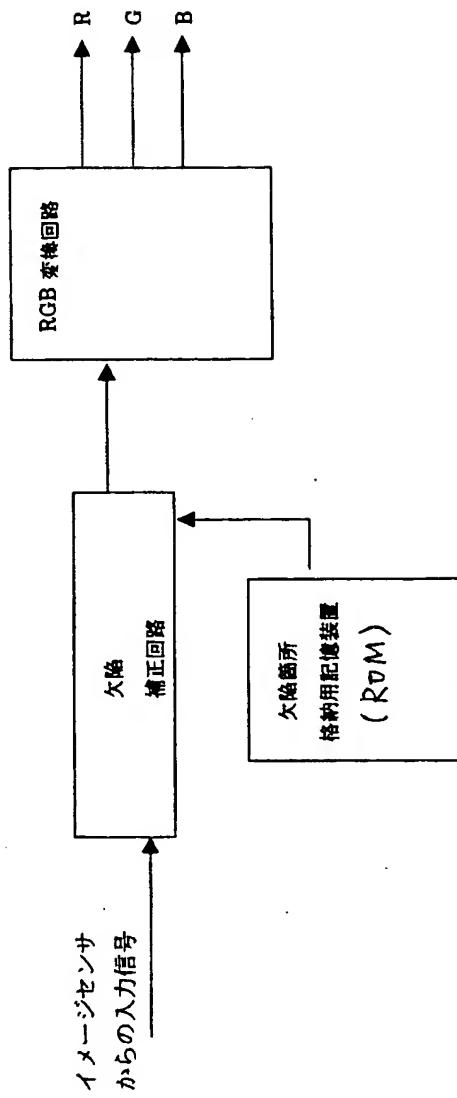
【図14】



【図15】



【図16】



【図17】

R Gr R Gr R Gr

Gb B Gb B Gb B

R Gr R Gr R Gr

Gb B Gb B Gb B

Rのタイミングで Gain1 offset1

Grのタイミングで Gain2 offset2

Gbのタイミングで Gain3 offset3

Bのタイミングで Gain4 offset4

【図18】

6×6画素を1ブロックとして考える。

R <sub>1,1</sub>	G <sub>2,1</sub>	R <sub>3,1</sub>	G <sub>4,1</sub>	R <sub>5,1</sub>	G <sub>6,1</sub>	R <sub>7,1</sub>	G <sub>8,1</sub>	R <sub>9,1</sub>	G <sub>10,1</sub>	R <sub>11,1</sub>	G <sub>12,1</sub>
G <sub>1,2</sub>	B <sub>2,2</sub>	G <sub>3,2</sub>	B <sub>4,2</sub>	G <sub>5,2</sub>	B <sub>6,2</sub>	G <sub>7,2</sub>	B <sub>8,2</sub>	G <sub>9,2</sub>	B <sub>10,2</sub>	G <sub>11,2</sub>	B <sub>12,2</sub>
R <sub>1,3</sub>	G <sub>2,3</sub>	R <sub>3,3</sub>		G <sub>4,3</sub>	R <sub>5,3</sub>	G <sub>6,3</sub>	R <sub>7,3</sub>	G <sub>8,3</sub>	R <sub>9,3</sub>		G <sub>10,3</sub>
G <sub>1,4</sub>	B <sub>2,4</sub>	G <sub>3,4</sub>		B <sub>4,4</sub>	G <sub>5,4</sub>	B <sub>6,4</sub>	G <sub>7,4</sub>	B <sub>8,4</sub>	G <sub>9,4</sub>		G <sub>11,4</sub>
R <sub>1,5</sub>	G <sub>2,5</sub>	R <sub>3,5</sub>	G <sub>4,5</sub>	R <sub>5,5</sub>	G <sub>6,5</sub>	R <sub>7,5</sub>	G <sub>8,5</sub>	R <sub>9,5</sub>	G <sub>10,5</sub>	R <sub>11,5</sub>	G <sub>12,5</sub>
G <sub>1,6</sub>	B <sub>2,6</sub>	G <sub>3,6</sub>	B <sub>4,6</sub>	G <sub>5,6</sub>	B <sub>6,6</sub>	G <sub>7,6</sub>	B <sub>8,6</sub>	G <sub>9,6</sub>	B <sub>10,6</sub>	G <sub>11,6</sub>	B <sub>12,6</sub>
R <sub>1,7</sub>	G <sub>2,7</sub>	R <sub>3,7</sub>	G <sub>4,7</sub>	R <sub>5,7</sub>	G <sub>6,7</sub>	R <sub>7,7</sub>	G <sub>8,7</sub>	R <sub>9,7</sub>	G <sub>10,7</sub>	R <sub>11,7</sub>	G <sub>12,7</sub>
G <sub>1,8</sub>	B <sub>2,8</sub>	G <sub>3,8</sub>	B <sub>4,8</sub>	G <sub>5,8</sub>	B <sub>6,8</sub>	G <sub>7,8</sub>	B <sub>8,8</sub>	G <sub>9,8</sub>	B <sub>10,8</sub>	G <sub>11,8</sub>	B <sub>12,8</sub>
R <sub>1,9</sub>	G <sub>2,9</sub>	R <sub>3,9</sub>		G <sub>4,9</sub>	R <sub>5,9</sub>	G <sub>6,9</sub>	R <sub>7,9</sub>	G <sub>8,9</sub>	R <sub>9,9</sub>		R <sub>11,9</sub>
G <sub>1,10</sub>	B <sub>2,10</sub>	G <sub>3,10</sub>		B <sub>4,10</sub>	G <sub>5,10</sub>	B <sub>6,10</sub>	G <sub>7,10</sub>	B <sub>8,10</sub>	G <sub>9,10</sub>		G <sub>11,10</sub>
R <sub>1,11</sub>	G <sub>2,11</sub>	R <sub>3,11</sub>	G <sub>4,11</sub>	R <sub>5,11</sub>	G <sub>6,11</sub>	R <sub>7,11</sub>	G <sub>8,11</sub>	R <sub>9,11</sub>	G <sub>10,11</sub>	R <sub>11,11</sub>	G <sub>12,11</sub>
G <sub>1,12</sub>	B <sub>2,12</sub>	G <sub>3,12</sub>	B <sub>4,12</sub>	G <sub>5,12</sub>	B <sub>6,12</sub>	G <sub>7,12</sub>	B <sub>8,12</sub>	G <sub>9,12</sub>	B <sub>10,12</sub>	G <sub>11,12</sub>	B <sub>12,12</sub>

6×6 ブロック内で4画素分(4個の計算式)の繰り返し



$$\begin{aligned}
 &= (R_{11} + R_{31} + R_{51} + R_{13} + R_{33} + R_{53} + R_{15} + R_{35} + R_{55}) / 9 \\
 &= (G_{21} + G_{41} + G_{61} + G_{23} + G_{43} + G_{63} + G_{25} + G_{45} + G_{65}) / 9 \\
 &= (G_{12} + G_{32} + G_{52} + G_{14} + G_{34} + G_{54} + G_{16} + G_{36} + G_{56}) / 9 \\
 &= (B_{22} + B_{42} + B_{62} + B_{24} + B_{44} + B_{64} + B_{26} + B_{46} + B_{66}) / 9
 \end{aligned}$$

同時処理方法は★ピクセルを例にすると、

$$R = (\Delta 1 + \Delta 2 + \Delta 3 + \Delta 4) / 4$$

$$G = (O A + O B + O C + O D) / 4$$

$$B = \square$$

【図19】

6×6画素を1ブロックとして考える。

R <sub>1,1</sub>	G <sub>2,1</sub>	R <sub>3,1</sub>	G <sub>4,1</sub>	R <sub>5,1</sub>	G <sub>6,1</sub>	R <sub>7,1</sub>	G <sub>8,1</sub>	R <sub>9,1</sub>	G <sub>10,1</sub>	R <sub>11,1</sub>	G <sub>12,1</sub>
G <sub>1,2</sub>	B <sub>2,2</sub>	G <sub>3,2</sub>	B <sub>4,2</sub>	G <sub>5,2</sub>	B <sub>6,2</sub>	G <sub>7,2</sub>	B <sub>8,2</sub>	G <sub>9,2</sub>	B <sub>10,2</sub>	G <sub>11,2</sub>	B <sub>12,2</sub>
R <sub>1,3</sub>	G <sub>2,3</sub>	I <sub>3,3</sub>	G <sub>4,3</sub>	I <sub>5,3</sub>	G <sub>6,3</sub>	R <sub>7,3</sub>	G <sub>8,3</sub>	I <sub>9,3</sub>	G <sub>10,3</sub>	I <sub>11,3</sub>	G <sub>12,3</sub>
G <sub>1,4</sub>	B <sub>2,4</sub>	G <sub>3,4</sub>	B <sub>4,4</sub>	G <sub>5,4</sub>	B <sub>6,4</sub>	G <sub>7,4</sub>	B <sub>8,4</sub>	G <sub>9,4</sub>	B <sub>10,4</sub>	G <sub>11,4</sub>	B <sub>12,4</sub>
R <sub>1,5</sub>	G <sub>2,5</sub>	I <sub>3,5</sub>	G <sub>4,5</sub>	I <sub>5,5</sub>	G <sub>6,5</sub>	R <sub>7,5</sub>	G <sub>8,5</sub>	I <sub>9,5</sub>	G <sub>10,5</sub>	I <sub>11,5</sub>	G <sub>12,5</sub>
G <sub>1,6</sub>	B <sub>2,6</sub>	I <sub>3,6</sub>	B <sub>4,6</sub>	G <sub>5,6</sub>	I <sub>6,6</sub>	G <sub>7,6</sub>	B <sub>8,6</sub>	I <sub>9,6</sub>	B <sub>10,6</sub>	G <sub>11,6</sub>	I <sub>12,6</sub>
R <sub>1,7</sub>	G <sub>2,7</sub>	R <sub>3,7</sub>	G <sub>4,7</sub>	R <sub>5,7</sub>	G <sub>6,7</sub>	R <sub>7,7</sub>	G <sub>8,7</sub>	R <sub>9,7</sub>	G <sub>10,7</sub>	R <sub>11,7</sub>	G <sub>12,7</sub>
G <sub>1,8</sub>	B <sub>2,8</sub>	G <sub>3,8</sub>	B <sub>4,8</sub>	G <sub>5,8</sub>	B <sub>6,8</sub>	G <sub>7,8</sub>	B <sub>8,8</sub>	G <sub>9,8</sub>	B <sub>10,8</sub>	G <sub>11,8</sub>	B <sub>12,8</sub>
R <sub>1,9</sub>	G <sub>2,9</sub>	I <sub>3,9</sub>	G <sub>4,9</sub>	I <sub>6,9</sub>	G <sub>6,9</sub>	R <sub>7,9</sub>	G <sub>8,9</sub>	I <sub>9,9</sub>	G <sub>10,9</sub>	I <sub>11,9</sub>	G <sub>12,9</sub>
G <sub>1,10</sub>	B <sub>2,10</sub>	G <sub>3,10</sub>	B <sub>4,10</sub>	G <sub>5,10</sub>	B <sub>6,10</sub>	G <sub>7,10</sub>	B <sub>8,10</sub>	G <sub>9,10</sub>	B <sub>10,10</sub>	G <sub>11,10</sub>	B <sub>12,10</sub>
R <sub>1,11</sub>	G <sub>2,11</sub>	I <sub>3,11</sub>	G <sub>4,11</sub>	I <sub>5,11</sub>	G <sub>6,11</sub>	R <sub>7,11</sub>	G <sub>8,11</sub>	I <sub>9,11</sub>	G <sub>10,11</sub>	I <sub>11,11</sub>	G <sub>12,11</sub>
G <sub>1,12</sub>	B <sub>2,12</sub>	I <sub>3,12</sub>	B <sub>4,12</sub>	G <sub>5,12</sub>	I <sub>6,12</sub>	G <sub>7,12</sub>	B <sub>8,12</sub>	I <sub>9,12</sub>	B <sub>10,12</sub>	G <sub>11,12</sub>	I <sub>12,12</sub>

6×6 ブロック内で 16 画素分(16 個の計算式)の繰り返し



$$= (49 * R_{11} + 7 * R_{31} + 7 * R_{13} + 1 * R_{33}) / 64$$



$$= (7 * R_{31} + 1 * R_{33} + 49 * R_{51} + 7 * R_{53}) / 64$$



$$= (7 * R_{13} + 1 * R_{33} + 49 * R_{15} + 7 * R_{35}) / 64$$



$$= (1 * R_{33} + 7 * R_{53} + 7 * R_{35} + 49 * R_{55}) / 64$$



$$= (49 * G_{21} + 7 * G_{41} + 7 * G_{23} + 1 * G_{43}) / 64$$



$$= (7 * G_{41} + 1 * G_{43} + 49 * G_{61} + 7 * G_{63}) / 64$$



$$= (7 * G_{23} + 1 * G_{43} + 49 * G_{25} + 7 * G_{45}) / 64$$



$$= (1 * G_{43} + 7 * G_{63} + 7 * G_{45} + 49 * G_{65}) / 64$$



$$= (49 * G_{12} + 7 * G_{32} + 7 * G_{14} + 1 * G_{34}) / 64$$



$$= (7 * G_{32} + 1 * G_{34} + 49 * G_{52} + 7 * G_{54}) / 64$$



$$= (7 * G_{14} + 1 * G_{34} + 49 * G_{64} + 7 * G_{36}) / 64$$



$$= (1 * G_{34} + 7 * G_{54} + 7 * G_{36} + 49 * G_{56}) / 64$$



$$= (49 * B_{22} + 7 * B_{42} + 7 * B_{24} + 1 * B_{44}) / 64$$



$$= (7 * B_{42} + 1 * B_{44} + 49 * B_{62} + 7 * B_{64}) / 64$$

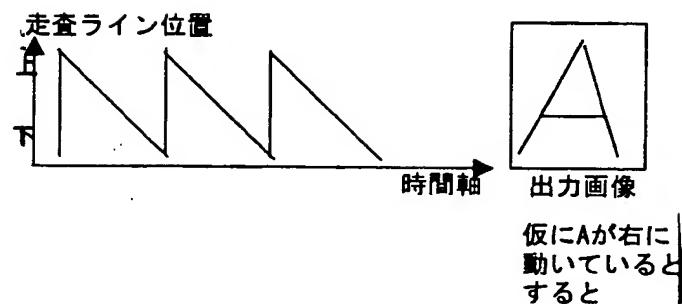


$$= (7 * B_{24} + 1 * B_{44} + 49 * B_{26} + 7 * B_{46}) / 64$$

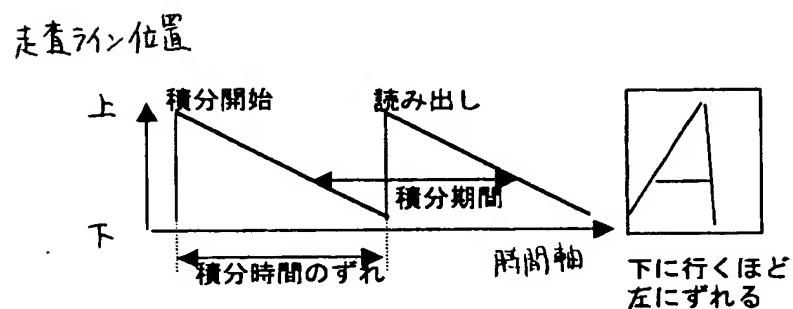


$$= (1 * B_{44} + 7 * B_{64} + 7 * B_{46} + 49 * B_{66}) / 64$$

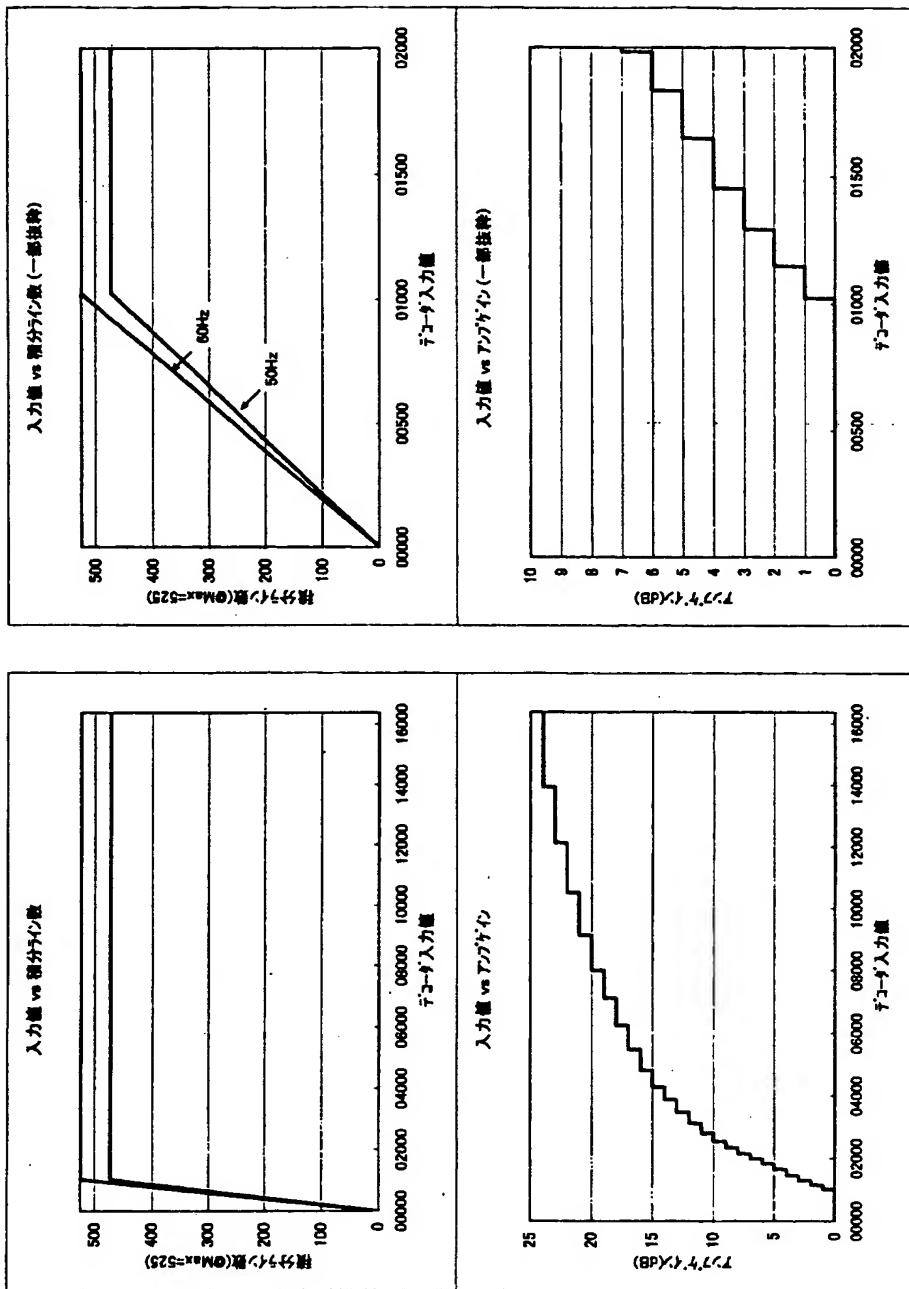
【図20】



【図21】



【図22】



【書類名】 要約書

【要約】

【目的】 本発明は、撮像された画像の画質を向上させたCMOSイメージセンサを提供することを提供することを目的とする。

【構成】 本発明のCMOSイメージセンサは、フォトダイオードと、少なくともリセットトランジスタ、増幅トランジスタ、水平方向選択トランジスタを備えたCMOSイメージセンサにおいて、リセットノイズの読み出し信号を監視し、任意の判定電圧に応じて信号をリセット電位に固定する検出回路を持つように構成される。

【選択図】 図1

出願人履歴情報

識別番号 [000005223]

1. 変更年月日 1996年 3月26日

[変更理由] 住所変更

住 所 神奈川県川崎市中原区上小田中4丁目1番1号  
氏 名 富士通株式会社